AMENDMENTS TO THE SPECIFICATION

CHANGE THE TITLE TO READ: --METHOD AND SYSTEM FOR DISPATCHING SERVICE REQUESTS TO SUB-APPLICATIONS--

Please replace the paragraph beginning on page 2, line 14 and ending on page 3, line 26 with the following amended paragraph marked up to show changes made relative to the immediate prior version:

A method and system for dispatching requests to sub-applications that use disparate logic models within the context of an application executing in a server environment is provided. The sub-applications form an application and share the same application context. The dispatching system allows sub-applications implemented with disparate logic models to share context information and thus function cooperatively as an application. The logic models may be the action-view model as described in U.S. Patent Application No. 09/753,037, entitled "Application Architecture," (Attorney Ref. No. 24376.8011US) filed on December 28, 2000 and the interaction model as described in U.S. Patent Application No. [[_____]] 09/681,567, entitled "Method and System for Executing a Computer Program," (Attorney Ref. No. 34570.8001US) (now U.S. Patent No. 6,944,851) filed concurrently with this application, both of which are hereby incorporated by reference. Other logic models may include workflow-based models. The term "logic model" refers to the overall architecture of a sub-application. In one embodiment, the dispatching system uses a common interface to interface with all sub-applications, regardless of their logic models. The common interface provides a service method or routine that the dispatching system invokes to effect processing by the sub-application. Each sub-application implements the common interface. In one embodiment, the dispatching system receives requests (e.g., HTTP requests), identifies the sub-applications that should process the received requests, and invokes the service routine of the identified sub-applications to process the received requests. Each sub-application may have an associated match criteria that indicates when the sub-application should process the requests. The dispatching system applies the match

criteria of the sub-applications to received requests to determine whether the associated sub-applications should process the requests. For example, a match criteria may be a regular expression (e.g., "*.html") that is applied to the uniform resource locator ("URL") of an HTTP request. The dispatching system processes the sub-applications in an order that may be predefined. The dispatching system selects the first sub-application and applies the match criteria associated with the first sub-application to the received request. If a match is found, then the dispatching system invokes the service routine of the first sub-application. When the first sub-application completes its processing, the dispatching system then selects the second sub-application and applies the matching criteria associated with the second sub-application to the received request. The dispatching system continues this process until all the sub-applications have been processed. In one embodiment, a sub-application may return an indication that the dispatching system should not process any additional sub-applications for the received request. For example, a sub-application that implements user authentication may prevent additional sub-applications from being processed when the authentication sub-application cannot authenticate the user who sent the request. The dispatching system may also stop processing additional sub-applications when a sub-application provides a response to the received request. In this way, application programs that are independently developed using different logic models can be combined to provide an overall application with the dispatching system invoking the service routine of the sub-applications as indicated by the match criteria. Moreover, legacy applications can be encapsulated within the common interface so that they can be combined with sub-applications that use different logic models.